Where's the joy?

Position Paper for "Using IT to Transform Unstructured, Creative Knowledge"
Panel Proposal to the 22nd Annual International Conference on Information Systems

Lynne P. Cooper April 11, 2001

When talking with information system professionals about knowledge management problems, they inevitably propose three solutions: build a website, build a database, and get a better search engine. While these IT-centric ideas generally have value as part of a complete solution, in and of themselves, they are insufficient, and often insulting.

In the innovative, creative, and often frenetic research & development world, a wide variety of knowledge is gathered, synthesized, tweaked, transformed, discarded, and rediscovered during the design process. The work consists of defining and solving problems in new ways. It's exciting and challenging work.

Some of the most well-regarded people in R&D communities function as "guru's": they have a personal, accessible base of knowledge founded on experience and theoretical understanding; they have large, calibrated networks of people that can provide advice; they have a repertoire of stories, anecdotes, and cases to illustrate their points; and they have the ability to step into the "seeker's" context and provide targeted information to help them. Working with a willing expert is a learning experience that is fast, efficient, and fun.

Many knowledge management systems attempt to fill in for guru's who either don't exist or aren't available. They can consist of large knowledge bases, warehouses of documents, profiling tools, expert systems, vast internet resources, and volumes of experts, personnel, and technology directories, all accessible through beyond-the-state-of-the-art search engines. Unlike the visitor to the guru, however, the seeker in this environment is relegated to listening to the village idiot spout out long lists of irrelevant material in the hopes that a gem may be revealed and recognized. It is tedious, frustrating, unproductive, and definitely not fun.

Knowledge management systems don't account for the joy that people feel in doing innovative work. Unlike a session with colleagues or a guru, the creative process is not invigorated by long stretches of hunting and browsing through websites and libraries. I see [at least] two ways in which knowledge management systems can improve their support of the innovation process. The first is to blend into the background. Like a good Victorian-era servant, they should be highly efficient, completely invisible and not demand attention from the user. Servant systems anticipate and act on the needs of the users in ways that require minimal involvement from the user. (Think Radar O'Reilly from M*A*S*H). The second is to be fun to use. Practioners in usability engineering are moving toward concepts such as "joy of use" (Glass, 1997) and hedonic quality (Hassenzahl, Beu, and Burmester, 2001) to describe the need for systems to meet the affective needs of the users. If a system is going to demand attention in order to be useful, then this system needs to be interesting and valuable enough to be a "partner" [or "spousal"?] system.

Neither of these are easy – especially if you limit yourself to IT-centric thinking. Whenever someone comes to me with an idea for a new database or website as a solution to an KMI-class problem, the first question I ask myself is whether it would be cheaper and better to hire someone and train them to do the job. Yes, that opens you up to the vulnerability of that person leaving, so then it becomes a trade between how many people to hire, train, and employ to provide the service vs. how many people to hire, train, and employ to build a tool. There's always a break even point that includes some human involvement to meet desired performance.

Note: this is the new technology part.

At the Jet Propulsion Laboratory, an institution that exists solely to be innovative, there are a number of knowledge management activities that cover the spectrum of solutions. The basic capabilities (e.g., high-speed internet access, portals, website hosting and registration, on-line directories, large scale document management systems, collaborative environments, a research library) exist as part of the infrastructure. In addition, there is research into content-heavy knowledge systems that are attempting to move into the servant/partner domains. The Technical Questions Database is an example of a "partner" type system. It

consists of a sets of questions in specific technical disciplines that would be applicable at a peer-review level. The database implementation is deceptively simple. The content, however, demands the attention of the users, who are responsible for browsing general categories, determining applicability to their domain, and responding to the questions during their normal design process.

A second research area, a "servant" type system, is the in-line capture of design information. Currently in a conceptual phase, the system is based on not interfering with the natural flow of the design session, and not demanding the attention of the participants to support the system. The capability is based on creating a transcript of design session discussions, analyzing the content, and producing products that are of immediate use to the design team (e.g., action item lists, decision map). In addition, information from other knowledge resources is offered to the design session based on a perceived need. The products produced during the design session are also available for knowledge mining activities that are looking beyond the needs of the individual product. The vision for this concept is to provide it as an inexpensive, real-time service available to any design team that wants it. It is currently being piloted as a skilled manual service that provides limited real-time products. If customer response warrants, emerging technology will be applied to improve the speed and level of automation of the service.

Moving beyond the general capabilities offered by knowledge management systems into a realm of servant/partner systems to support specific domains is challenging. It requires both an understanding of the natural work processes and the ways (human, computer, other) knowledge can be applied and managed to improve them. It is easier to build a website or database. It's more productive and satisfying to augment people's ability to do work. Though, of course, search engines are getting better every day....

Hassenzahl, M., Beu, A., and Burmester, M. (2001). Engineering joy. IEEE Software, January/February, pp. 70-76.

Glass, R. (1997). "The Looking Glass," www.sun.com.au/news/onsun/oct97/page6.html